DNS Summer Daysの チュートリアルの歩き方

2015年7月24日

DNS Summer Days 2015

株式会社日本レジストリサービス（JPRS）

平林有理

2015年7月27日更新

講師自己紹介

- 氏名：平林 有理(ひらばやし ゆうり)
- 生年月日：1990年12月31日（24歳）
- 所属：株式会社日本レジストリサービス（JPRS） システム部
- 略歴：
  - 2013年4月　大学院入学、DNSSEC/DANEと出会う
  - 2015年4月　JPRS入社
  - 2015年7月1日　JPRS システム部 配属
  - **2015年7月24日　DNS Summer Days 2015 講師**
- 実装・運用の経験は非常に浅いので、本日は、みなさんと共に学んでいけたらと思います

本日のお話の対象となる方、目的、内容

- 対象となる方
  - DNSサーバーを今後、運用される初学技術者の方
  - すでに運用されている方の知識のおさらい
- 目的
  - DNSを学ぶ上で鍵となる知識をお持ち帰りいただくこと
- お話しする内容
  - 2012年～2014年の期間にDNS Summer Daysで発表されたチュートリアルを体系的に整理し、そのポイントを説明する
- お話しない内容
  - DNS Summer Daysチュートリアルで扱われていない監視の運用設計、評価などは参考資料の紹介にとどめる

目次

- 過去のチュートリアルの分類分け
- 過去のチュートリアルのポイント
  - 成り立ちと概要
  - 仕様
  - システム設計
  - 設定
  - 運用
- まとめ
- 参考資料紹介

過去のチュートリアルの分類わけ

発表者	タイトル	発表年	ジャンル
森下 泰宏	DNS入門	2012	成り立ち
滝澤 隆史	DNS再入門	2014	仕様
滝澤 隆史	DNSのRFCの歩き方	2012	
山口 崇徳	DNSのシステム設計	2013	設計
高嶋 隆一	DNS設定例の紹介(オーソリティティブ)	2014	設定
山口 崇徳	DNS設定例の紹介(キャッシュ)	2014	
東 大亮	DNSキャッシュサーバの設定ノウハウ	2014	
水野 貴史	初心者のためのDNS運用入門	2014	運用
伊藤 高一	DNSのよくある間違い	2012	
山口 崇徳	DNSトラブルシューティング	2012	
森下 泰宏	教科書には載っていないDNS	2013	仕様(応用)

アップデートが存在する資料は、最新版のみ記載

注意

- 本発表は、過去の発表資料を引用する形での紹介を行っています
- 詳しい内容については元の資料を参照してください

成り立ちと概要

タイトル	DNS入門
話者	森下 泰宏 - 株式会社日本レジストリサービス
発表	DNS Summer Days 2012
資料URL	http://dnsops.jp/event/20120831/20120831-DNS_Summer_Days_2012-DNSprimer-v1.2.pdf
概要	DNSの成り立ちと基本動作の詳細
目次	• HOSTS.TXTからドメイン名・DNSまでの道のり • DNSの基本構造と名前解決の基本動作 • 構造に由来するDNSの美点・弱点と弱点克服のためのさまざまな工夫 • 持って生まれた悲しい宿命とそれに立ち向かうための必要事項

本日紹介しない範囲について

仕様から運用まで一通り学んだあと、DNSの弱点に対する様々な工夫など、より深いDNSの成り立ちや構造を理解したいときに参照ください

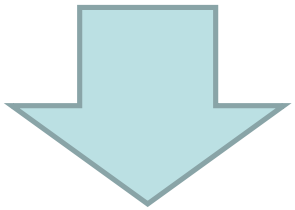
赤字部分をピックアップしてお話します

インターネットにおける通信のしくみ

- インターネットにおける通信では、IPアドレスという番号で相手を指定・識別している
  - 送信側：受信側のIPアドレスを指定してデータを送る
  - 受信側：送信元のIPアドレスにより、どの相手からデータが届いたのか識別する

名前とIPアドレスの対応付け

- IPアドレスは人間が記憶するには大変
  - IPv4（ex：192.0.2.1）
  - IPv6（ex：2001:db8:10:8f01:face:b00c:0:25）
  - IPアドレスは変化する可能性もある



- IPアドレスよりも記憶しやすく使いやすい名前を使う
  - 名前とIPアドレスを対応付ける何らかのしくみが必要

基本的な2つの方法

- それぞれユーザーが個別にデータベースを作り使用する
  - 携帯電話の電話帳機能と同等

- 一つのデータベースをみんなで共有する
  - サーバーにデータベースのファイルをおいておき、ユーザーに配布する

名前の一意性を確保するにはデータベースの共有が必要

すべての名前がインターネット全体で同じ意味を持つこと

一つのデータベースを共有

- DNSができる前は、データーベースファイルの共有という形で運用されていた
  - データベースはHOSTS.TXTという名前で、SRI-NICという団体により集中管理・公開
  - この名前の名残はUNIXやWindowsなどに残っている
    - UNIX　　/etc/hosts
    - Windows　　C:¥Windows¥System32¥drivers¥etc¥hosts

HOSTS.TXT方式の破綻

- インターネットの成長により登録されているコンピューターの数が増え、うまく機能しなくなっていった

- SRI-NICの負荷増大
  - HOSTS.TXTの巨大化、更新頻度の増加
- ネットワークの負荷増大
  - HOSTS.TXTを取得するユーザーの増加
- ユーザーの負荷増大
  - 最新版の入手・設定・再配布の必要性

DNSの登場

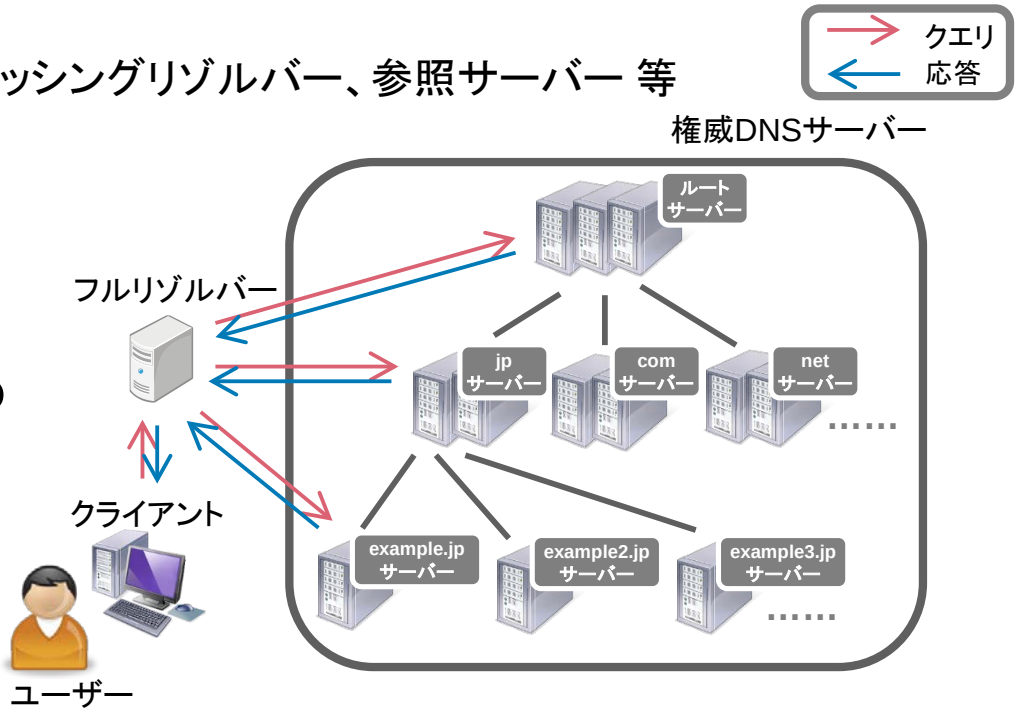
- DNS：Domain Name System
  - ドメイン名（Domain Name）を使えるようにするために開発されたシステム（System）

- ドメイン名に対応させる形でデータベースを分散
  - 担当する部分のデータベースをそれぞれが管理
  - ✓ 負荷の分散
- 分散管理されたデータベースをネットワークで共有
  - 全体を1つのデータベースのように見せる
  - ✓ HOSTS.TXTと同様に名前の一意性を確保

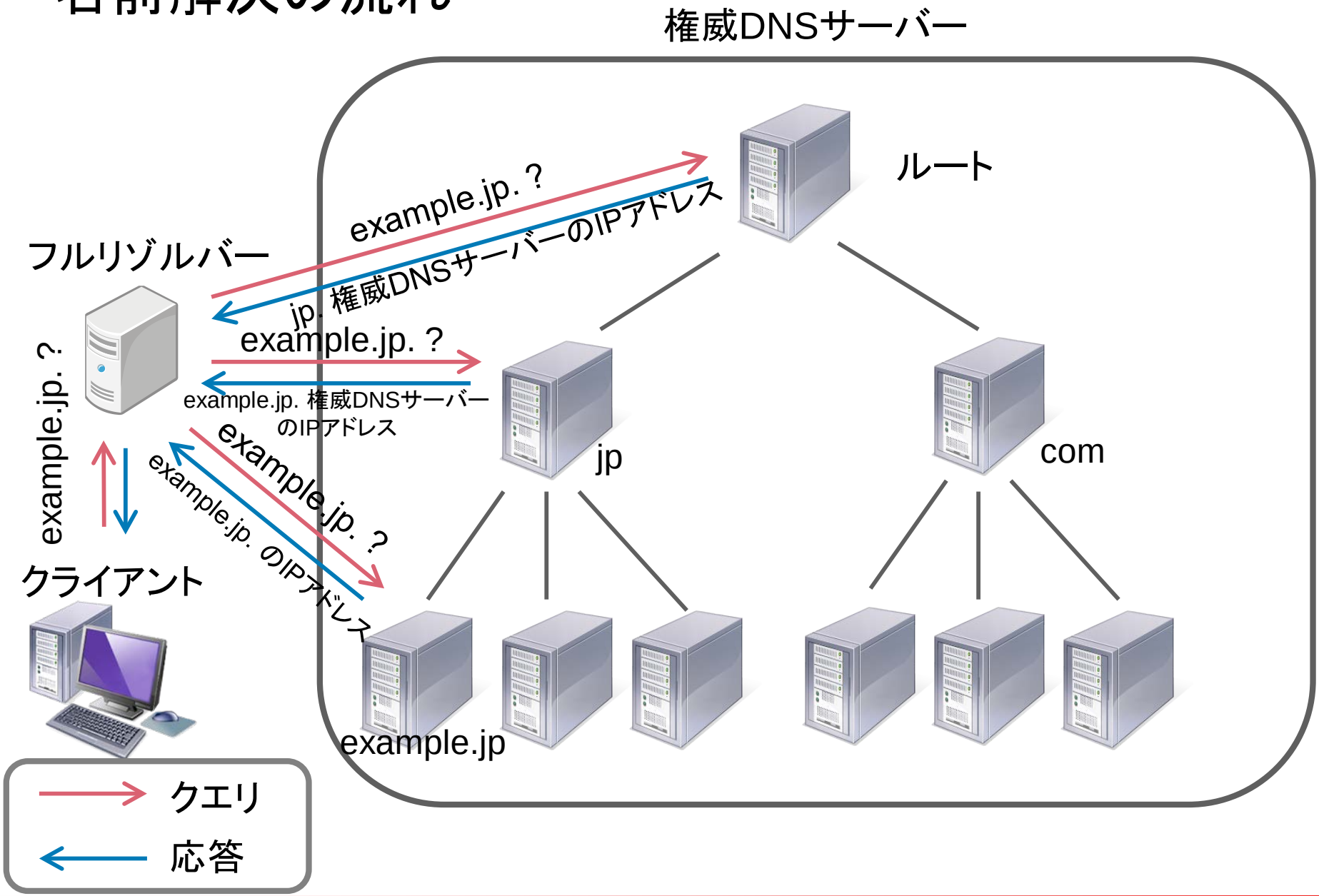
2種類のDNSサーバー

- 階層構造を構成するサーバー（分散管理）
  - 権威DNSサーバー
    - 権威サーバー、DNSコンテンツサーバー 等
- 階層構造をたどるサーバー（名前解決）
  - フルリゾルバー
    - キャッシュDNSサーバー、キャッシングリゾルバー、参照サーバー 等

という 2つの役割を持つ
DNSサーバーが存在する



名前解決の流れ



仕様

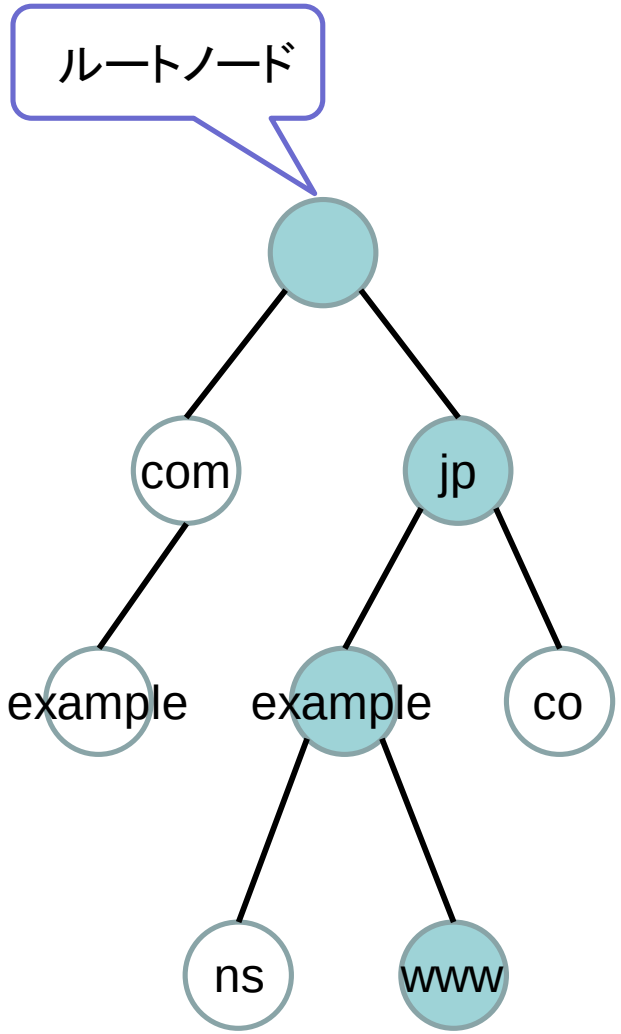
タイトル	DNS再入門
話者	滝澤 隆史 - 株式会社ハートビーツ
発表	DNS Summer Days 2013, 2014
資料URL	http://dnsops.jp/event/20140626/DNS-primer.pdf
概要	DNSの仕様の教科書
目次	• DNSの背景 • DNSの概要 • ドメイン名 • ドメイン名の管理 • リソースレコード • マスターファイル • DNSメッセージ • リゾルバとネームサーバ

本日紹介しない範囲について
実際の運用において、仕様の詳細を確認・理解したいときに参照ください

赤字部分をピックアップしてお話します

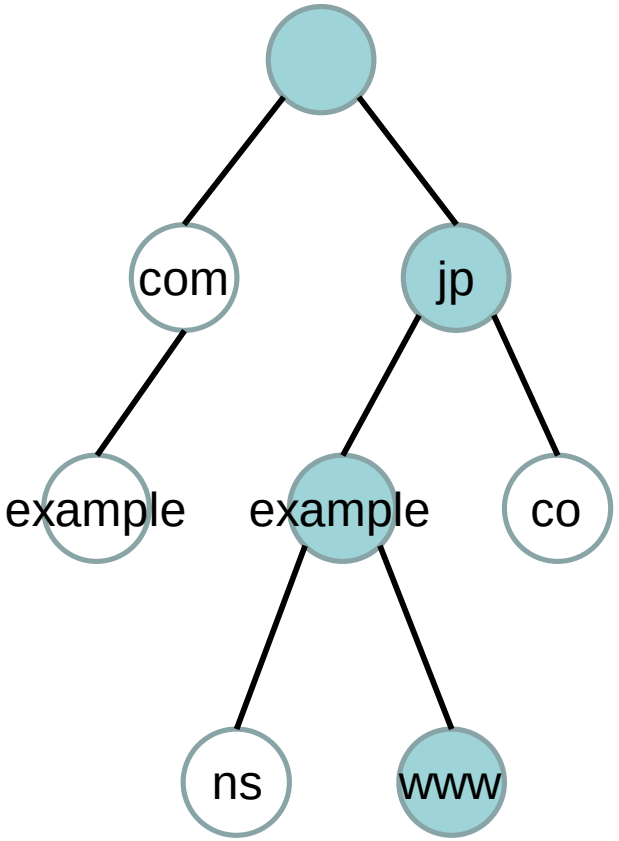
ドメイン名の構造

- ドメイン名空間はツリー構造になっている
- 各ノードはラベルを持つ
  - ルートノードのためにnullラベルが予約されている
- ノードのドメイン名はそのノードからルートノードまでのラベルのリストになっている
  - ex) “www” “example” “jp” “(null)”



絶対ドメイン名と相対ドメイン名

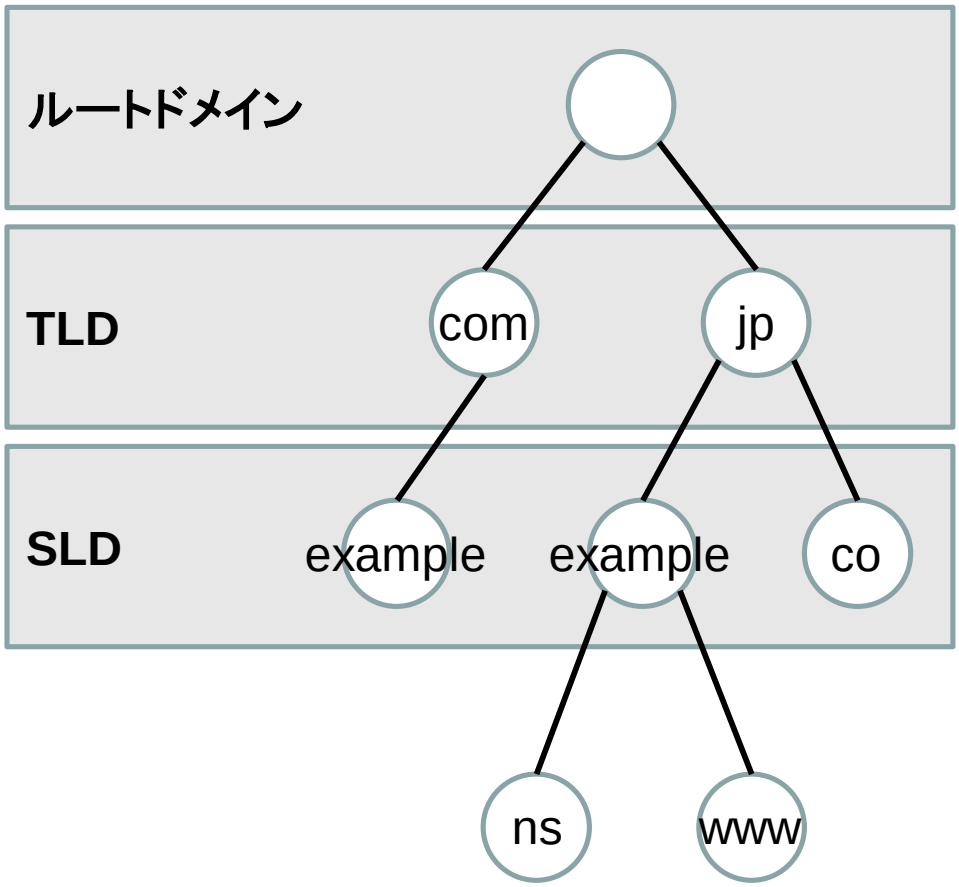
- 絶対ドメイン名
  - ドットで終わるドメイン名
  - ex) “www.example.jp.”
- 相対ドメイン名
  - 親のドメイン名に対して相対的に表したドメイン名
  - ex) “www”は“example.jp.”の相対ドメイン名



ルートドメイン、TLD、SLD

- 各ノードはノードの深さによって名前がつく

- ルートドメイン
- TLD
  - トップレベルドメイン
- SLD, 2LD
  - セカンドレベルドメイン



検索リスト

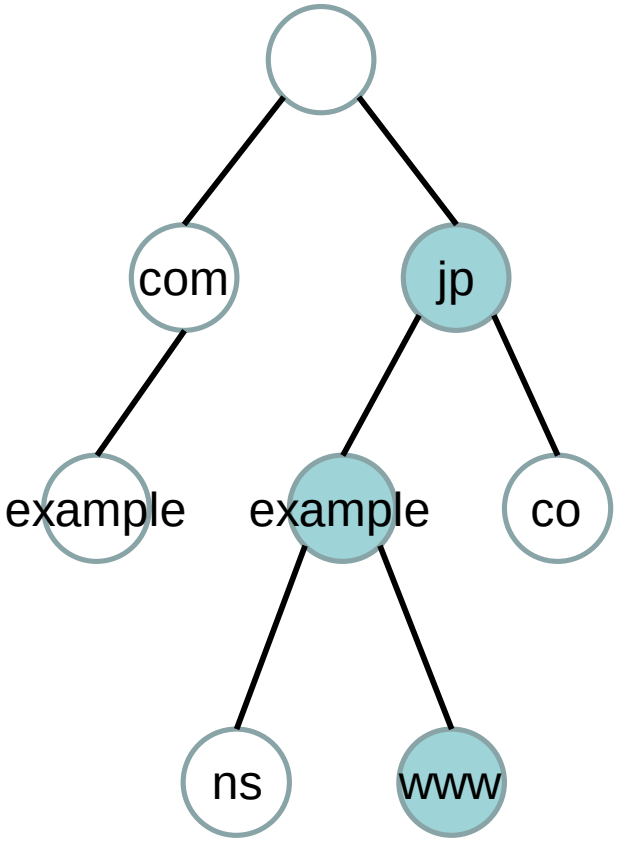
- 相対ドメイン名に親ドメイン名を補完する際のドメイン名のリスト
  - /etc/resolv.confの “domain” と “search”

/etc/resolv.confの例

```
domain example.jp
nameserver 192.0.2.1
nameserver 192.0.2.2
```

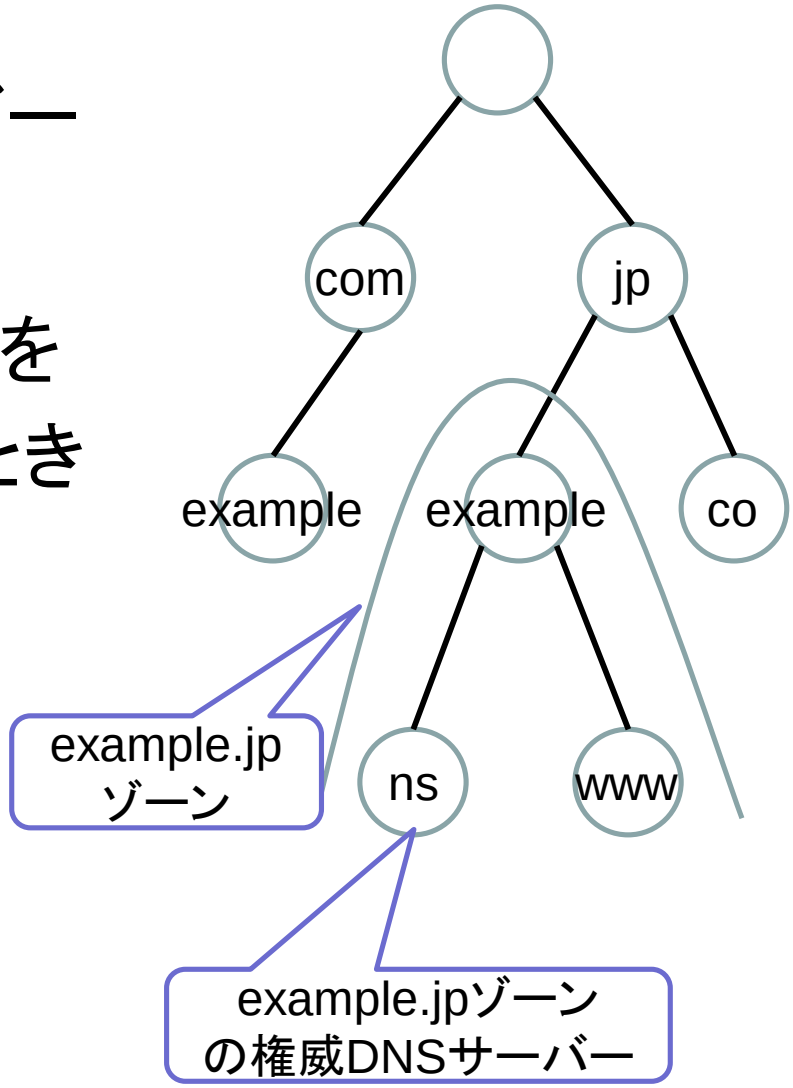
完全修飾ドメイン名(FQDN)

- TLDまでのラベルを含んだドメイン名を完全修飾ドメイン名と呼ぶ
  - FQDN(Fully Qualified Domain Name)
- ソフトウェアがドメイン名を扱うときは基本的にFQDNを用いる
- FQDNはルートドメイン名の相対ドメイン名と考えても良い
  - 検索リストのメンバーとしてルート“.(null)”が解釈されるため



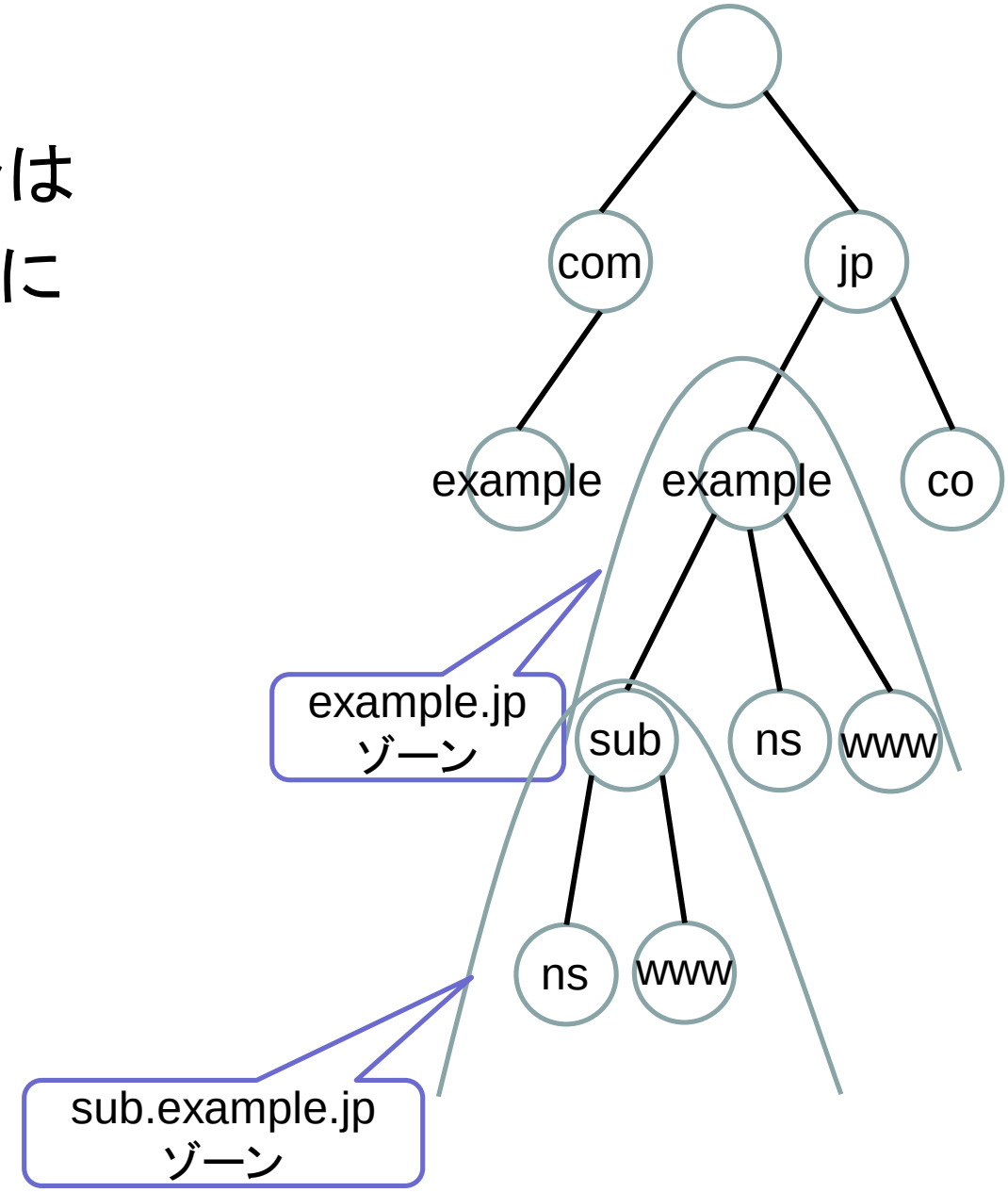
ゾーンと権威

- ドメイン名を管理する単位をゾーンと呼ぶ
- ネームサーバーがそのゾーンを管理できる権限を持っているときそのゾーンの権威となる



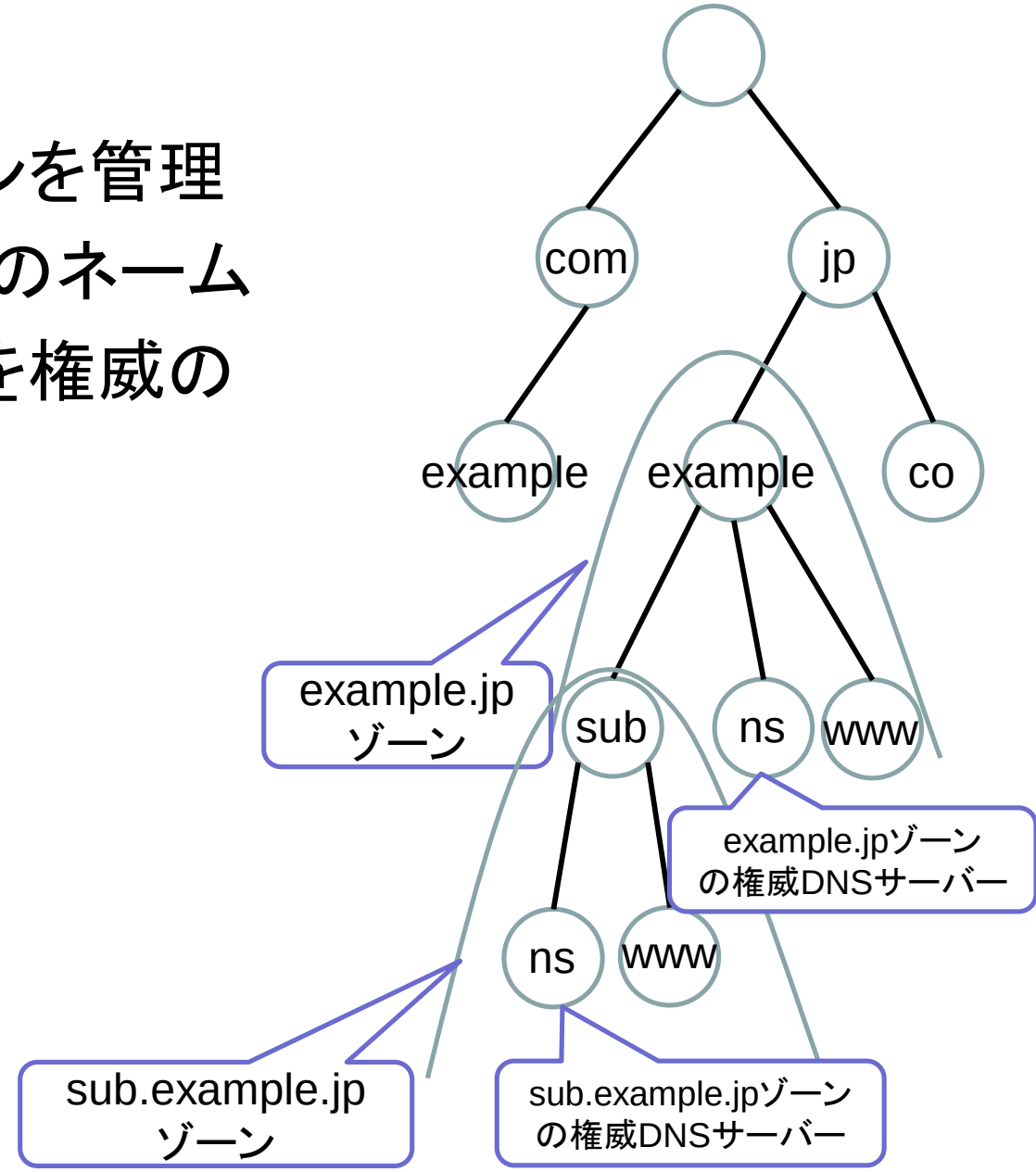
ゾーンの分割

- 各ドメイン名のゾーンは
  サブドメインのゾーンに
  分割することが可能



権威の委任

- この分割されたゾーンを管理する正式な権限を他のネームサーバに委せることを権威の委任と呼ぶ



タイトル	DNSのRFCの歩き方
話者	滝澤 隆史 - 株式会社ハートビーツ
発表	DNS Summer Days 2012
資料URL	http://dnsops.jp/event/20120831/DNS-RFC-PRIMER-2.pdf
概要	RFCの中でDNSはどのように規定されているか

目次	• RFCの読み方
	• DNSの基本仕様
	  • RFC1034の概要
	  • ドメイン名空間とリソースレコード
	  • ネームサーバー
	  • リゾルバー
	  • RFC1035の概要
	  • ドメイン名とリソースレコードの実装
	  • メッセージ
	  • マスターファイル
	  • 実装
	• アップデートRFC

本日紹介しない範囲について

DNSの開発を行う際や、運用上の問題に遭遇したとき、本来の仕様を確認する際に参考になります

赤字部分をピックアップしてお話します

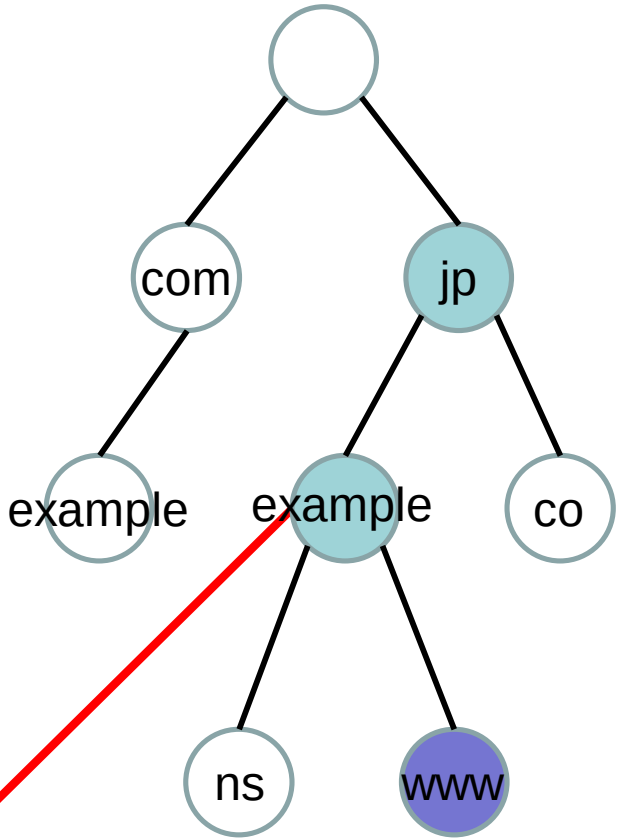
DNSの基本仕様のRFC

- RFC 1034
  - DNSの構成要素の役割や機能についての説明
- RFC 1035
  - RFC 1034で定めた役割、機能を実現するためのドメイン名システムとプロトコルについての詳細を記述

- 注意点
  - 作成された当時と現在では時代背景が異なる
    - DNSが検討されたのはARPANETからThe Internetへの過渡期
  - 曖昧さや、間違いがある
    - 後に発行されたRFCによってアップデートされている

RFC1034 – 3.6 リソースレコード

- 各ノードはリソース情報の集まりをもつ
  - 空でもよい
- 特定の名前に関連付けられたリソース情報の集まりは別々のリソースレコード(RRs)から構成される
- 集まりの中のRRsの順番は指定できないし、維持される必要もない



リソースレコード

```
example.jp.     IN SOA ns.example.jp. …
example.jp.     IN NS  ns.example.jp.
ns.example.jp.  IN A   192.0.2.1
```

RFC1034 – 3.6 リソースレコード
リソースレコードの用語

- owner
  - そのRRがあるドメイン名
- TTL
  - RRが破棄されるまでキャッシュしても良い期間を示す秒単位32bitの値
- class
  - プロトコルファミリーを識別する符号化された16bitの値
    - IN (the INternet system), CH(the CHaos system)
- type
  - このRRのリソースのタイプを識別する符号化された16bitの値
    - SOA, NS, A, AAAA, MX, CNAME, PTR, TXT など
- RDATA
  - タイプとクラスに依存するデータ

RFC1034 – 3.6.1 RRsのテキスト表現

- RRは一行で示される。複数行になる場合は括弧を使う

```
ns1.a.example.   IN      A       192.0.2.54

@                IN      SOA     ns1.a.example.  root.localhost. (
                                 1047 604800 86400 2419200 3600
                                 )
```

- 行の先頭はRRのowner

```
www.a.example.   IN      A       192.0.2.58
```

- 空白で始まる行はownerが前のRRと同じと想定

```
mail.a.example. IN      A       192.0.2.57
.....................   IN      AAAA    2001:db8:53::25
```

システム設計

タイトル	DNSのシステム設計
話者	山口 崇徳 - 株式会社インターネットイニシアティブ
発表	DNS Summer Days 2013
資料URL	http://dnsops.jp/event/20130719/20130719-dns-design-yamaguchi-2.pdf
概要	運用開始後には変更が難しいシステムの設計方法

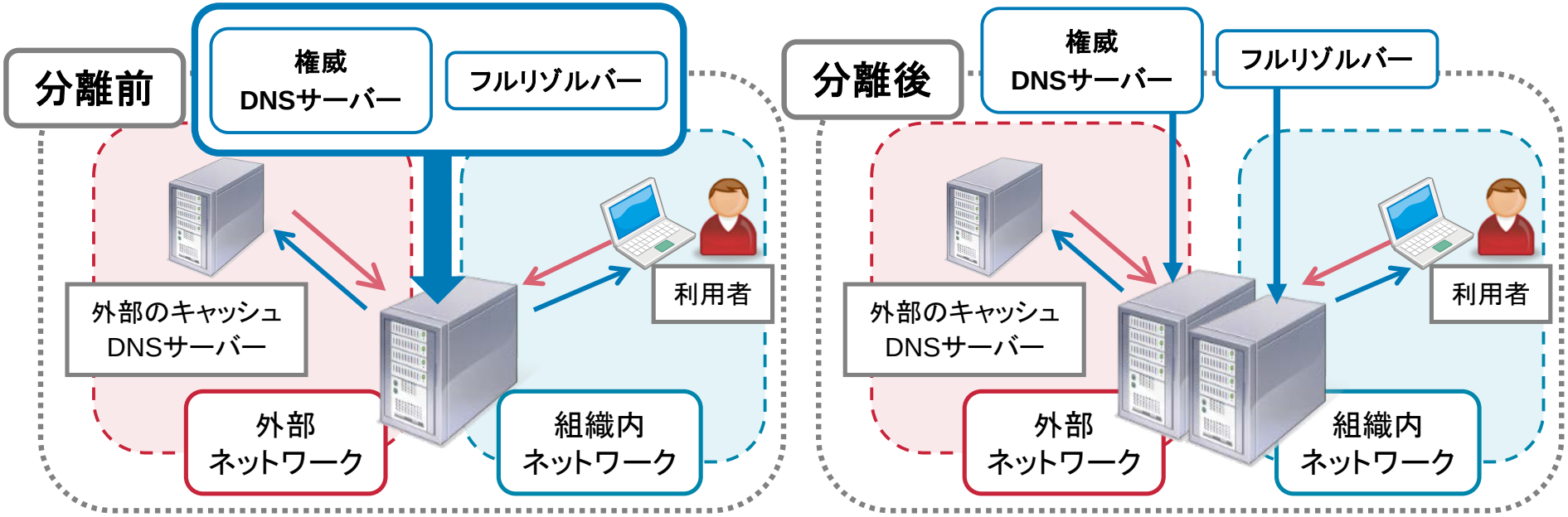
目次	• DNS設計の基本
	  • 2種類のDNSの役割分担
	  • DNSサーバのハードウェア
	  • ネットワーク構成
	• 権威サーバの設計
	  • 名前空間の設計
	  • 権威サーバの構成
	• 参照サーバの設計
	  • 参照サーバのIPアドレス
	  • resolve.confの更新タイミング
	• 応用

本日紹介しない範囲について
DNSのシステム設計を行う上で疑問が生じた際に参照ください

赤字部分をピックアップしてお話します

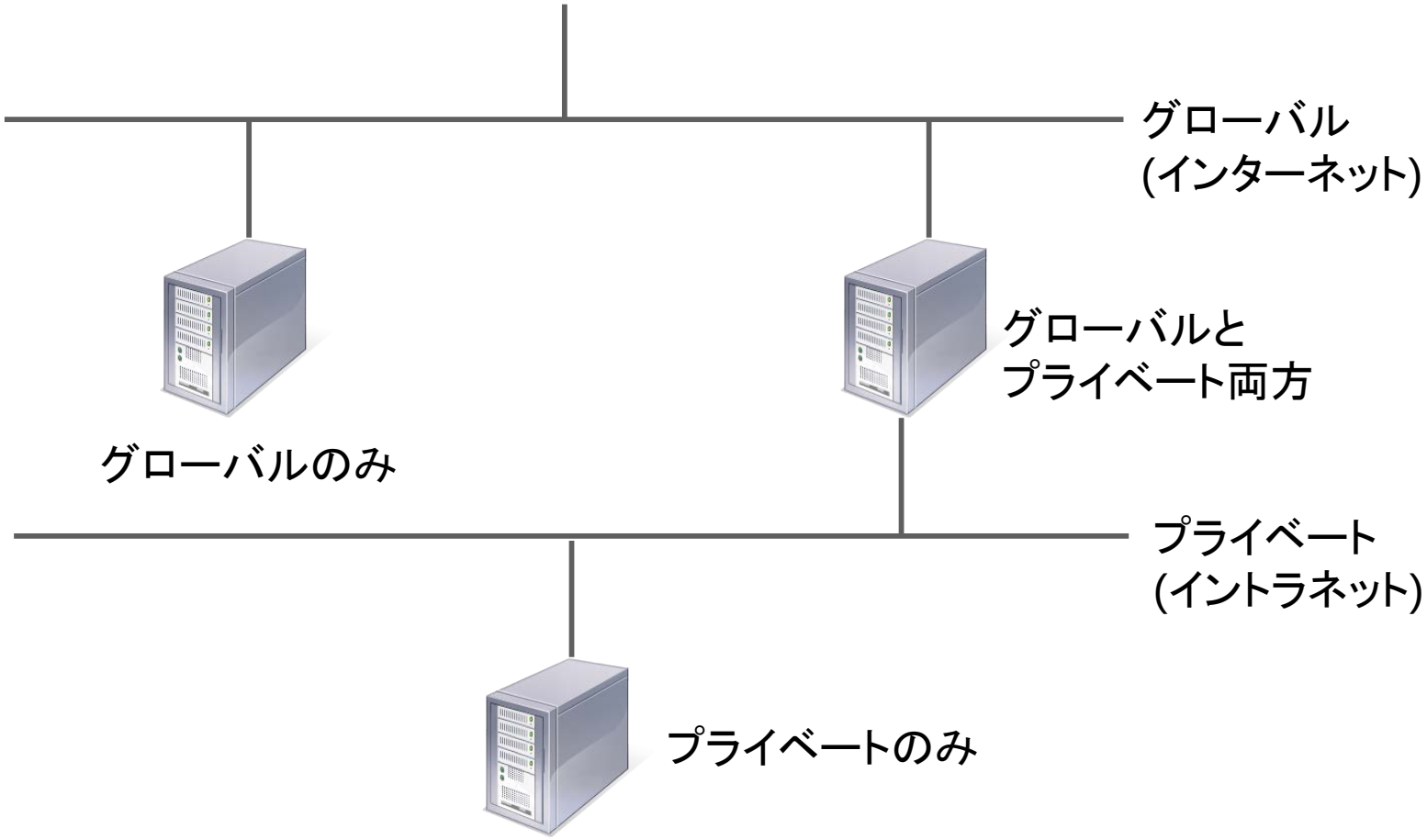
2種類のDNSサーバーの役割分担

- 権威DNSサーバーとフルリゾルバーは、同じDNSプロトコルを扱うサーバーだが、役割がまったく異なる
- 2つの機能を混在させることでDNSキャッシュポイズニング攻撃の被害を受ける可能性が上がる



ネットワーク構成

- ネットワーク上のどこにサーバーを設置するか



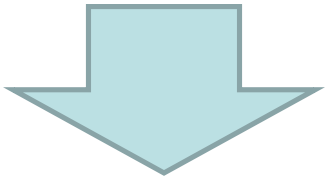
ネットワーク構成

	グローバル	プライベート
外部用権威DNSサーバー	✓	
内部用権威DNSサーバー		✓
フルリゾルバー	✓	✓

- フルリゾルバーにグローバルIPアドレスをもたせる運用
  - グローバル側からのアクセスには十分考慮する必要がある
  - NAT変換を行っての運用も可能だがNAT変換テーブルあふれなどに注意

名前空間の設計

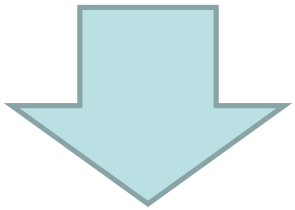
- どのような名前をつけるか
  - http://www.example.jp　　or　　http://example.jp
  - http://www.example.jp/foo　　or　　http://foo.example.jp
- キャンペーンサイトなどを本サイトのサブドメインで運用するか?新規でドメイン名を登録するか?
  - キャンペーン終了後ドメイン名をどのように扱うか?
- どうやって管理するか?



- どのように行うのかは、それぞれの運用ポリシーによる
  - まずは、運用ポリシーを決める必要がある

フルリゾルバーのIPアドレス

- DNSで名前解決はできるが、フルリゾルバー自身の名前解決はできない
  - フルリゾルバーはIPアドレスで直接指定する必要がある
- DNS設定をクライアントに配布する仕組み(DHCP, IPCPなど)は存在するが…
  - DHCPを無視するクライアントの存在



- 一度公開したフルリゾルバーのIPアドレスは変更できないものと考え、ネットワークを設計する必要がある

設定

タイトル	DNS設定例の紹介(オーソリティティブ)
話者	高嶋 隆一 - DNSOPS.JP
発表	DNS Summer Days 2014
資料URL	http://dnsops.jp/event/20140626/20140626-DNS-SD-Ryuichi.pdf
概要	“ns1.dnsops.jp”, “urquell.酔っ払い.jp”で運用実績がある BIND 9 設定例

目次	• BIND 9(named.conf)の設定例
	  • options{}
	  • logging{}
	  • zone{}
	  • 便利設定
	  • その他共通設定

本日紹介しない範囲について
権威DNSサーバーの発展的な設定が必要な際、参照ください

赤字部分をピックアップしてお話します

ns1.dnsops.jpのBIND 9 設定紹介(option)

```
options
{
        directory       “/var/named”; // the default
        dump-file       "data/cache_dump.db";
        statistics-file         "data/named_stats.txt";
        memstatistics-file      "data/named_mem_stats.txt”;
};
```

BIND9の設定ファイルの親パスとrndcの出力先の設定

ns1.dnsops.jpのBIND 9 設定紹介(logging)

```
logging
{
        channel default_debug{
                file            "data/named.run";
                severity        dynamic;
                print-category yes;
                print-severity yes;
                print-time     yes;
                };
        channel default_channel{
                file    "/var/log/named.log“ size 10M versions 10;
                severity        dynamic;
                print-category yes;
                print-severity yes;
                print-time     yes;
                };
```

logファイルの表示設定

10世代までログを残し、一つ一つのログファイルは10MBまで

ns1.dnsops.jpのBIND 9 設定紹介(logging)

```
category queries { default_debug; };

category update-security { default_channel; };
category default { default_channel; };
category general { default_channel; };
category database {default_channel; };
category security { default_channel; };
category config { default_channel; };
category resolver { default_channel; };
category notify { default_channel; };
category client { default_channel; };
category unmatched { default_channel; };
category network { default_channel; };
category update { default_channel; };
category query-errors { default_channel; };
category dispatch { default_channel; };
category dnssec { default_channel; };
category delegation-only { default_channel; };
category edns-disabled { default_channel; };
```

クエリ関係のログのみ
default_debug
あとは
default_channelへ

logのカテゴリ設定

ns1.dnsops.jpのBIND 9 設定紹介(logging)

```
channel xfer_channel {
file “/var/log/named-xfer.log” size 10M versions 10;
severity dynamic;
print-category yes;
print-severity yes;
print-time yes;
};
category xfer-in { xfer_channel; };
category xfer-out { xfer_channel; };
```

ゾーン転送に関するログは別ファイルへ

ns1.dnsops.jpのBIND 9 設定紹介(zone)

```
zone "dnsops.jp“ {
        type    master;
        file    "dnsops.jp.signed";
        allow-transfer {183.181.160.83; };
        notify yes;
};
zone "dnssec.jp“ {
        type    master;
        file    "dnssec.jp.signed";
        allow-transfer {183.181.160.83; };
        notify yes;
};
```

allow-transferでslaveサーバにのみゾーン転送を許可

タイトル	DNS設定例の紹介(キャッシュ)
話者	山口 崇徳 - 株式会社インターネットイニシアティブ
発表	DNS Summer Days 2014
資料URL	http://dnsops.jp/event/20140626/cache-config.pdf
概要	フルリゾルバーのセキュリティ設定と他DNSとの連係動作

目次	• unboundのアクセス制限
	  • オープンリゾルバ
	  • NATとポートランダム
	• 他DNSサーバとの連携
	• 設定例

本日紹介しない範囲について
フォワーダなど他のDNSサーバとの連携動作、プライベートゾーンでの運用方法などを理解する際に参考になります

赤字部分をピックアップしてお話します

なぜアクセス制限するのか？

- フルリゾルバーはセキュリティ的な被害を受けやすい
- DNSキャッシュポイズニング攻撃
  - 権威DNSサーバからの応答に偽の応答を割り込ませることでユーザーを悪意のあるサイトに誘導する攻撃
  - アクセス制限することで
    - 攻撃がしづらくなる
    - 攻撃者から攻撃が成功したかの観測が困難になる
- DNS amp 攻撃の踏み台
  - アドレスを詐称したクエリによって、別のアドレスへDNS応答を仕向け帯域を飽和させる攻撃
  - 被害者となるだけでなく加害者となってしまう可能性
  - アクセス制限することで影響が限定的になる

アクセス制限

- Unboundのアクセス制限例(ホスト内のクエリのみ許可)

```
access-control: 0.0.0.0/0 refuse
access-control: 127.0.0.0/8 allow
```

- マッチしたクライアントに対する挙動

allow	アクセス許可(非再帰クエリは拒否)
allow_snoop	アクセス許可(非再帰も許可)
deny	クエリを破棄(応答を返さない)
refuse	クエリを拒否(拒否応答を返す)

- ローカルネットワークから以外のクエリを拒否または破棄することを推奨

DNSキャッシュポイズニングの被害を防ぐために

- 絶対してはいけない設定(BIND)

```
query-source port 53;
```

- この設定をすることでソースポートランダマイゼーションが無効になり、ソースポートが53に固定される
- 問い合わせソースポートの固定はDNSキャッシュポイズニング攻撃の成功率を著しく高める
  - ランダム　　1 / 43億
  - 固定　　1 / 6.5万
- Unboundでは特に意識することなくソースポートランダマイゼーションを利用可能

NATとソースポートランダマイゼーション

- NAT(NAPT)の中でフルリゾルバーを運用すれば外からの偽の応答は届かない？
  - ポート番号が的中した場合NAT変換されてフルリゾルバーに応答が到達する

- 一部のNAT機器ではソースポートを外部から推測しやすい値に変換することがあり、注意が必要

タイトル	DNSキャッシュサーバの設定ノウハウ
話者	東 大亮
発表	DNS Summer Days 2014
資料URL	http://dnsops.jp/event/20140626/DNS-design-operation-higashi_final.pdf
概要	パフォーマンスチューニングとフルリゾルバーを運用する上で考慮すべきトラブル

目次	• パフォーマンスチューニング
	• トラブルを避ける設計と運用
	  • IPフラグメントが届かない問題
	  • TCPに対応しないクライアントの問題
	• トラブルへの備え
	• DNSキャッシュサーバの監視
	• セキュリティについて
	  • dns-0x20

本日紹介しない範囲について

チューニング設定によってフルリゾルバー内の動作がどのように変化するのか、DNS応答が大きいときにどのような問題が起こるのか、図を使った詳解があります

赤字部分をピックアップしてお話します

パフォーマンスチューニング

- クライアントから受信した未解決の再帰検索要求の処理状態を管理・保持する領域のサイズを引き上げ

BIND 9 のデフォルト値

```
recursive-clients 1000
```

Unboundのデフォルト値

```
num-queries-per-threads 512 or 1024
```

  - デフォルトは数千QPS以上のフルリゾルバーでは小さすぎる

- キャッシュメモリのサイズ

BIND 9 のデフォルト値

```
max-cache-size 制限無し
```

  - デフォルトではシステムのメモリを食い尽くしてしまう恐れがある

Unboundのデフォルト値

```
rrset-cache-size: 4MB
msg-cache-size:4MB
```

  - デフォルトでは多数のクライアントを収容するには小さすぎる

トラブルを避ける設計と運用 - DNS応答が大きい場合に起こる問題

- IPフラグメントが届かない問題
  - UDP応答がフラグメント化されてフルリゾルバーに送信され、これにより、途中のネットワーク経路上に問題があると、IPフラグメントが疎通できず応答が受け取れないことがある
- TCPに対応しないクライアントの問題
  - EDNS0が無効かつDNS応答が512byteを超える場合にTCPが使われる
  - クライアントの中にはTCPに対応せず512byteを超える応答が扱えないものが存在する

根本的な解決にはクライアント側の対応が必要

minimal-responsesオプション

- BINDのminimal-responsesオプションを利用することでDNS応答サイズを小さくすることが可能

設定なし

```
% dig @localhost jprs.co.jp

; <<>> DiG 9.10.0-P1 <<>> @localhost jprs.co.jp
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 19999
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 3, ADDITIONAL: 6

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;jprs.co.jp.                    IN      A

;; ANSWER SECTION:
jprs.co.jp.             86400   IN      A       202.11.16.167

;; AUTHORITY SECTION:
jprs.co.jp.             86400   IN      NS      ns1.jprs.co.jp.
jprs.co.jp.             86400   IN      NS      ns2.jprs.co.jp.
jprs.co.jp.             86400   IN      NS      ns3.jprs.co.jp.

;; ADDITIONAL SECTION:
ns1.jprs.co.jp.         86400   IN      A       202.11.16.49
ns1.jprs.co.jp.         86400   IN      AAAA    2001:df0:8::a153
ns2.jprs.co.jp.         86400   IN      A       202.11.16.59
ns2.jprs.co.jp.         86400   IN      AAAA    2001:df0:8::a253
ns3.jprs.co.jp.         86400   IN      A       61.200.83.204

;; Query time: 934 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Fri Jun 20 00:40:21 JST 2014
;; MSG SIZE  rcvd: 213
```

設定あり

```
% dig @localhost jprs.co.jp

; <<>> DiG 9.10.0-P1 <<>> @localhost jprs.co.jp
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 27868
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;jprs.co.jp.                    IN      A

;; ANSWER SECTION:
jprs.co.jp.             86400   IN      A       202.11.16.167

;; Query time: 238 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Fri Jun 20 00:49:54 JST 2014
;; MSG SIZE  rcvd: 55
```

AUTHORITY SECTION
ADDITIONAL SECTION
がない

運用

タイトル	初心者のためのDNS運用入門
話者	水野 貴史 - 株式会社日本レジストリサービス
発表	DNS Summer Days 2013, 2014
資料URL	http://dnsops.jp/event/20140626/dns-beginners-guide2014-mizuno.pdf
概要	トラブルシューティングの基本とツールの使い方

目次	• DNSトラブルシューティングの基本
	  • 区別すべき2種類の問い合わせ
	• 道具の使い方
	  • コマンドラインツールの使い方
	  • Webサービスの紹介
	• よくあるトラブル事例とトラブルシューティング
	  • 名前が引けないときの調査
	  • 名前を引くのに時間がかかるときの調査
	  • シリアルの変更ミスと解決法

本日紹介しない範囲について

トラブルシューティングでのdig実践的利用方法を知ることができます

赤字部分をピックアップしてお話します

トラブルシューティングに有用なツール

- フルリゾルバーの挙動をたどる
  - dig
  - drill

- 全体を俯瞰する
  - Squish.net DNS traversal checker
  - dnscheck.jp

digコマンドとは

- DNSサーバーにクエリを送り、応答を調査するコマンド
  - リクエストに関するパラメーターを細かく調整して、応答を調査できる
  - BINDに付属
  - Unobundに付属のdrillコマンドもほぼ同等の機能を備えている

```
$ dig␣+rec␣@192.0.2.53␣example.jp.␣SOA
```

オプション　DNSサーバー　対象ドメイン名　クエリタイプ

調査に使えるWebサービス

- DNSの設定などを、GUIで可視化・チェック可能
- Squish.net DNS traversal checker（個人提供：James氏）
  - http://dns.squish.net
  - DNS可視化ツール
  - 応答のおかしいDNSサーバーなどを調べることが可能
- dnscheck.jp（提供:JPRS）
  - http://dnscheck.jp
  - DNSの設定チェックツール
  - 今現在の設定の確認

タイトル	DNSトラブルシューティング
話者	山口 崇徳 - 株式会社インターネットイニシアティブ
発表	DNS Summer Days 2012
資料URL	http://dnsops.jp/event/20120831/dns-troubleshoot-2.pdf
概要	トラブルシューティング例と解決方法

目次	• ツールの紹介
	• 参照サーバのトラブル
	  • キャッシュの消し方
	  • resolv.conf読み込みのタイミング
	• 権威サーバのトラブル
	  • シリアル番号上げ損ね
	  • lame delegation
	  • プライベートアドレスの逆引き
	  • CNAME関連
	• 実在するドメインの問題調査
	• クライアント側のトラブル

本日紹介しない範囲について
トラブルの実践的な切り分け方法を学ぶ際、参考になります

赤字部分をピックアップしてお話します

フルリゾルバーのトラブル - 古いキャッシュのクリア

- 古いキャッシュが残っているために、名前解決に失敗する場合、キャッシュをクリアすることで解決できる場合がある
  - 権威DNSサーバー側の設定が間違っている場合、古いキャッシュが残っているためにアクセスできている場合があることを考慮する

```
BIND
$ rndc flushname <対象のキャッシュname>
$ rndc flushtree <対象のキャッシュname> (9.9)
Unbound
$ unbound-control flush <対象のキャッシュname>
$ unbound-control flush_type <対象のキャッシュname> <type>
$ unbound-control flush_zone <対象のキャッシュname>
```

非推奨（すべてのキャッシュがクリアされる）

```
BIND
$ rndc flush
Unbound
$ unbound-control reload
```

フルリゾルバーのトラブル - resolv.confの変更反映

- resolv.confを修正したけど変更前のアドレスへ問い合わせする
  - 名前解決のたびにresolv.confが読み込まれるわけではない
  - resolv.confの読み込みはプロセス起動直後の初期化時のみ
  - 再初期化しないと変更は反映されない
  - 再初期化するにはプロセスの再起動が必要

- これに気がつかず変更前のフルリゾルバーを停止すると名前解決できなくなる

権威DNSサーバーのトラブル - SOAシリアルの上げ損ね

- ゾーンを更新！シリアルをあげよう！
  - “YYYYMMDDnn”形式を使うルールで運用
  - “2015072401”のつもりが”2150072401”に！
  - シリアルは加算しないと更新できない！

- “YYYYMMDDnn”形式での運用をやめるしかない…？

シリアル巻き戻しテクニックの利用(RFC1982 Serial Number Arithmeic)

次のスライドで解説

シリアル巻き戻しテクニック(RFC1982)

1. 上げそこなったシリアルに2^32-1(=2147483647)を加算した値をセット
   - ex) 2150072401 + 2147483647 = 4297556048
2. スレーブへの反映を確認
   - dig +norec @[SLAVE] [DOMAIN] SOA
3. 目的のシリアル値をセット
   - ex) 2015072401
4. スレーブへの反映を確認

タイトル	**DNSのよくある間違い**
話者	伊藤 高一 - 株式会社ブロードバンドタワー
発表	DNS Summer Days 2012
資料URL	http://dnsops.jp/event/20120831/kohi-p1.pdf
概要	設定例でこうなっているからで流しがちな間違いを紹介

目次	• DNSアーキテクチャ
	• SOAレコードのおさらい
	  • シリアル更新ミスと解決法
	• ゾーン転送の概要とトラブルシューティング
	• lame delegation
	• ゾーンデータの表記方法
	  • 記述失敗例
	  • CNAMEのしてはいけないこと・しないほうがよいこと

本日紹介しない範囲について
運用前に間違った設定、認識をしていないかというチェックリストとして参照ください

赤字部分をピックアップしてお話します

ゾーンファイルの記述ミス

- 名前の末尾にピリオドを忘れると…

```
$ORIGIN          a.example.
@                IN      SOA     ns1.a.example.  root.localhost. (…)
IN       NS      ns1.a.example.
IN       MX      10 mail.a.example
```

```
$ORIGIN          a.example.
@                IN      SOA     ns1.a.example.  root.localhost. (…)
                 IN      NS      ns1.a.example.
                 IN      MX   10 mail.a.example.a.example.
```

- これを防ぐには設定ファイル表記の流儀を決めておく
  - 相対表記は使わない
  - ownerは必ず相対表記、RDATAは必ず絶対表記

  など

CNAMEでしてはいけないこと

- CNAMEを定義したownerに対して他のRRを定義してはいけない

```
www.example.    IN      CNAME           www1.example.
                        MX      10      mx.example.
```

- NSやMXのRDATAに、CNAMEで定義したaliasを書いてはいけない

```
$ORIGIN         a.example.
@               IN      NS              ns
ns                      CNAME           example.test.
```

CNAMEでしないほうがよいこと

- CNAMEのCNAME(多段CNAME)

```
alias1      IN    CNAME       alias2
alias2            CNAME       alias3
alias3            CNAME       alias4
```

  - 循環参照の元
- RFCでは規定がないため、何段まで動作するかは実装による
  - BINDは16段
  - Unboundは8段
- こちらの権威DNSサーバだけでなく相手のフルリゾルバーにも依存する

まとめ

ここまでの総まとめ

- HOSTS.TXTの弱点を克服するために、誕生したDNS！
- 一度運用を始めるとなかなか設定を変えられない部分についてポリシーをよく検討！
- digやWebサービスを有効活用！
- オープンリゾルバはダメ、絶対！
- 基本をマスターしたら応用的な資料を！
- 実際にDNSサーバーを動かして色々試してみよう！

参考資料紹介

仕様(応用)

タイトル	教科書には載っていないDNS
話者	森下 泰宏 - 株式会社日本レジストリサービス
発表	DNS Summer Days 2013
資料URL	http://dnsops.jp/event/20130719/20130719-undocumented-DNS-orange-6.pdf
概要	「DNS入門」で省略した委任の仕組みと詳細
目次	• グルーと内部名・外部名 • 委任応答とreferral • グルーはグルー（DNSデータのランキング） • カミンスキー型攻撃手法

本資料について

委任の仕組みやDNSに対するセキュリティ攻撃の手法を知り、対策を講じたいとき参考になる資料です。

評価(1)

タイトル	**DNSの評価と計測の話**
話者	服部 成浩 - SCSK株式会社
発表	Internet Week 2013
資料URL	https://www.nic.ad.jp/ja/materials/iw/2013/proceedings/d2/d2-hattori.pdf
概要	DNSストレスツールの使い方と評価方法

目次	• DNSストレスツール
	  • dnsperf と resperf の違い
	  • 負荷の生成方法
	  • エラーメッセージ対処
	• ストレスツール実例
	  • ケーススタディ

本資料について
DNSの負荷に対する評価方法を検討する際、参考になる資料です

評価(2)

タイトル	DNSとメール
話者	安高 元気 - 楽天株式会社
発表	Internet Week 2013
資料URL	https://www.nic.ad.jp/ja/materials/iw/2013/proceedings/d2/d2-yasutaka.pdf
概要	送信ドメイン認証によって発生する権威DNSサーバ・フルリゾルバーの負荷と対応について

目次	• 送信ドメイン認証の考え方とその仕組み
	  • 送信ドメイン認証に関連する主な技術
	• 権威DNS サーバへのクエリと負荷
	  • 送信ドメイン認証/DKIM の普及状況
	  • キャッシュDNS サーバへのクエリと負荷
	• MTA からキャッシュDNS サーバへのクエリ
	  • (Case 2)鍵長が「長く」なることによる影響
	  • (Case 2)鍵長のサイズが512byte超えることによる影響

本資料について

DNS応答の肥大化によって権威DNSサーバ・フルリゾルバーの負荷がどうなるのか知りたいとき参考になる資料です

評価(3)

タイトル	**JP DNSへのRRLの導入**
話者	阿波連 良尚 - 株式会社日本レジストリサービス
発表	Internet Week 2013
資料URL	https://www.nic.ad.jp/ja/materials/iw/2013/proceedings/d2/d2-aharen.pdf
概要	JP DNSへDNS RRLを適用するまでの評価の流れ

目次	• DNSリフレクター攻撃の概要と対策
	• JP DNSサーバーへのDNS RRLの導入
	• 評価のステップ
	  • 机上評価
	  • 社内評価
	  • フィールド評価
	  • 実運用への投入

本資料について

システムの評価どのように進めていくか、ステップごとの評価方法について参考となる資料です

監視

タイトル	権威DNSの監視
話者	坂口 智哉 - 株式会社日本レジストリサービス
発表	Internet Week 2014
資料URL	https://www.nic.ad.jp/ja/materials/iw/2014/proceedings/d1/d1-sakaguchi.pdf
概要	権威DNSサーバーならではの監視
目次	• ゾーン抽出の監視 • ゾーン転送の監視 • JP DNS監視 • 監視で重要なこと

本資料について
権威DNSサーバーを監視する際に考慮すべき点について参考になる資料です

JPRS
JAPAN REGISTRY SERVICES